



#5

[Partial Translation]

JAPANESE LAID-OPEN PATENT APPLICATION NUMBER 62-42237

Application Date August 20, 1985

Laid Open on February 24, 1987

RECEIVED

AUG 16 1989

TC 2700 MAIL ROOM

ABSTRACT OF THE DISCLOSURE

An upper program counter 2 (hereafter referred to as the UPC 2) is a counter specifying an address for a batch of data to be read together from memory. A lower program counter 3 (hereafter referred to as the LPC 3) is a counter showing the position of a start instruction in the batch of data. If, for example, the size of each piece of data in the memory is four bytes and the size of a unit in an instruction buffer is two bytes, the UPC 2 is expressed by 2^2 bits or more and the LPC 3 limited to a 2^1 -bit value.

When the value of the LPC 3 is 0, the value of a write pointer 4 is advanced each time a piece of read data the size of a unit in the instruction buffer is written in the instruction buffer.

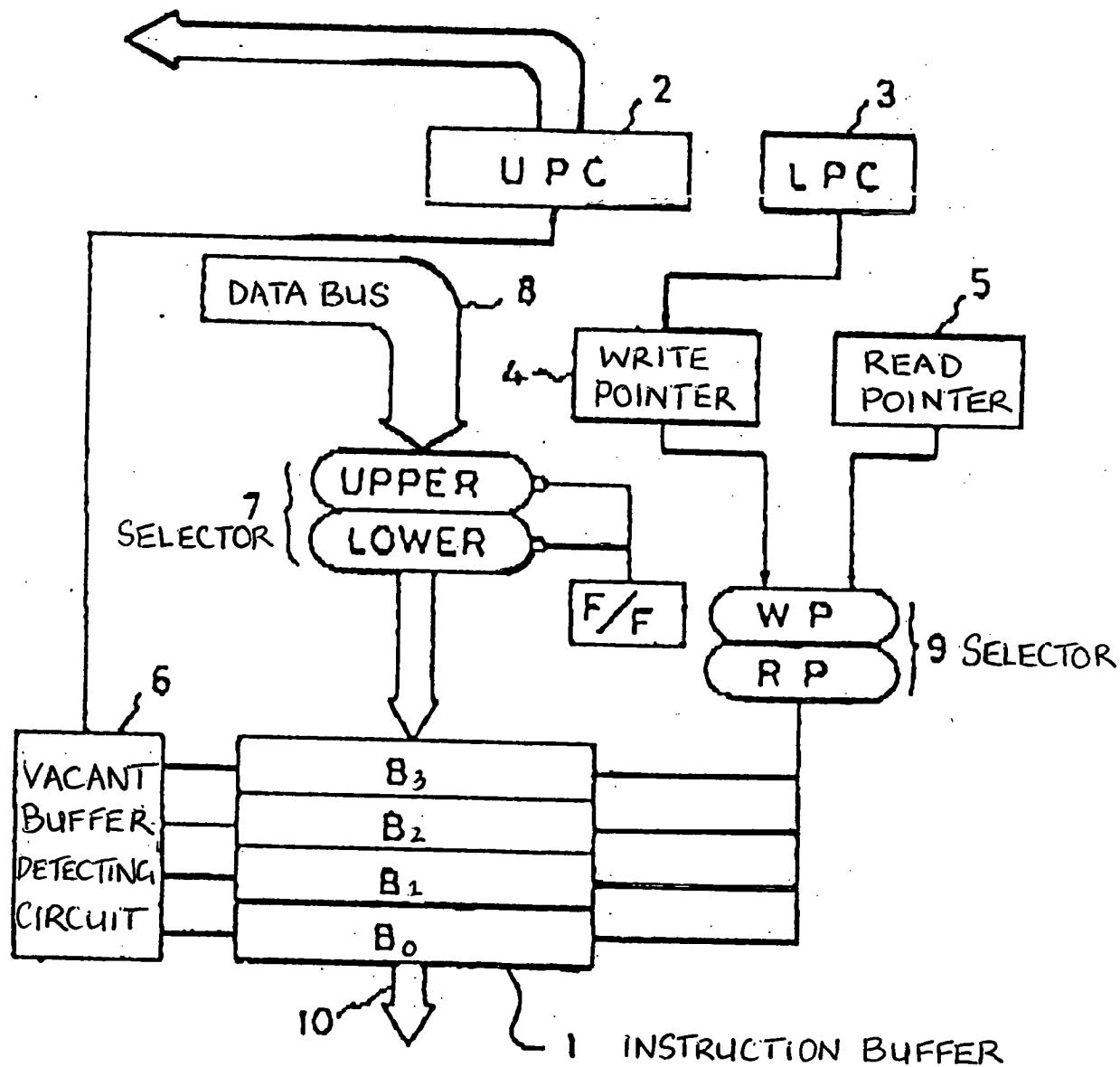
When the value of the LPC 3 is other than 0, it is reduced by one each time a piece of read data the size of a unit in the instruction buffer is written in the instruction buffer. The position indicated by the write pointer 4 does not change from the start of the instruction buffer. This process

is repeated until the value of the LPC 3 changes to 0.

As a result of the above process, the start instruction of a batch of data will always be set at the start of the instruction buffer regardless of its position within the read data that is read in one batch.

Selected Drawing: Fig. 1

FIG 1



⑩ 公開特許公報 (A) 昭62-42237

⑪ Int.CI.
G 06 F 9/38
9/32

識別記号 庁内整理番号
A-7361-5B
7361-5B

⑫ 公開 昭和62年(1987)2月24日

審査請求 未請求 発明の数 1 (全4頁)

⑬ 発明の名称 命令バッファへのロード方式

⑭ 特願 昭60-180926
⑮ 出願 昭60(1985)8月20日

⑯ 発明者 右手 秀樹 青梅市末広町2丁目9番地 株式会社東芝青梅工場内
⑰ 出願人 株式会社東芝 川崎市幸区堀川町72番地
⑱ 代理人 弁理士 本田 崇

明細書

1. 発明の名称

命令バッファへのロード方式

2. 特許請求の範囲

夫々の欄が命令長の単位幅となっている複数個のバッファ部から成る命令バッファに、前記単位幅よりも長いデータをメモリから一度に読み出してロードする命令バッファへのロード方式において、前記メモリからデータを読み出したときその読み出されたデータ中における一の命令の先頭位置を示す数値がセットされ、以降その読み出されたデータが前記単位幅のデータ毎に順次前記命令バッファに格納されるか否かカウントダウンするカウンタを設け、前記メモリからデータを読み出したとき、前記カウンタにセットされた数値がその読み出されたデータの先頭位置を示す所定値であればその読み出されたデータを前記単位幅毎に前記バッファ部の第1番目から順次格納し、前記カウンタにセットされた数値が前記所定値でないときはその数値が前記所定値となるまで前記バッフ

ア部の第1番目のみを書き換えてその読み出されたデータを前記単位幅毎に格納し、その数値が前記所定値となった後は前記バッファ部の第2番目からその読み出されたデータの残りの部分を前記単位幅毎に順次格納することを特徴とする命令バッファへのロード方式。

3. 発明の詳細な説明

【発明の技術分野】

本発明は、メモリに格納されている命令を命令バッファにロードする命令バッファへのロード方式に関する。

【発明の技術的背景】

従来、命令バッファへのロードは各命令ごとに行われ、必ず命令バッファの先頭に命令の先頭が位置するようになっていた。これは、メモリから読み出すデータの幅が小さいため、プログラムカウンタによって指定されるアドレスに格納されたデータを読み出せば必ずそのデータの先頭が命令の先頭と一致するからである。

【背景技術の問題点】

しかし、メモリに対して一度読み出させるデータ幅が拡張されるに伴い、命令の先頭がメモリから読み出したデータの途中にある場合が生じてきた。このため、新たに命令フェッチをやり直す場合、命令パッファの幅が読み出したデータの幅に比べて小さい構成では命令パッファの先頭部分に不要データが残存する場合が生じる。このような場合、従来では、この不要データを命令パッファから追い出す操作を別途行うか、また、予めメモリ上のデータの中から有効な部分の長さを検出し、その部分のみを読み出す操作を行わなければならなかった。このような操作は命令フェッチの速度を低下させる原因となっている。

【発明の目的】

本発明は上記欠点に鑑みなされたもので、その目的は、メモリ内でアクセスできるデータ幅が命令パッファの幅よりも大きい場合、命令をロードしたときに命令パッファの先頭にメモリから読み出したデータ中の命令の先頭を一致させる操作（命令の頭出し操作）を簡単なハードウェアで実

の部分を上記単位幅毎次格納するようにして上記の目的を達成している。

【発明の実施例】

以下、図面を参照して本発明の一実施例を説明する。

第1図は、本発明方式を適用する取扱いられる装置の構成ブロック図である。図中1は命令パッファである。命令パッファ1は命令の単位幅（最小変化幅ともいう）を幅とする4個のパッファ部から成る。夫々のパッファ部をその先頭からB₀・B₁・B₂・B₃とする。2, 3はメモリ（図示せず）に格納されているデータのアドレスをセットするためのプログラムカウンタである。そのうち2はメモリ上のアドレスを指定する部分であり、メモリ内のデータのアクセス単位に応じて、そのアドレスの下位ビットが切り捨てられた形でセットされるもので、これを上位プログラムカウンタ（以下UPCと略す）という。一方、3は、読み出すべき命令の先頭がメモリから一度に読み出されるデータのどの位置にあるかを示すカウン

タし、常にメモリに対しては、メモリ内のデータ幅に応じた読み出しを行ってだけで命令の頭出しを可能にすることである。

【発明の概要】

そこで本発明では、メモリからデータを読み出したときその読み出されたデータ中ににおける一の命令の先頭位置を示す数値がセットされ、以後その読み出されたデータが上記単位幅のデータ毎に順次前記命令パッファに格納される都度カウントダウンするカウンタを設け、上記メモリからデータを読み出したとき、上記カウンタにセットされた数値がその読み出されたデータの先頭位置を示す所定値であればその読み出されたデータを上記単位幅毎に上記パッファ部の第1番目から順次格納し、上記カウンタにセットされた数値が上記所定値でないときはその数値が前記所定値となるまで上記パッファ部の第1番目のみを書き換えてその読み出されたデータを上記単位幅毎に格納し、その数値が上記所定値となった後は上記パッファ部の第2番目からその読み出されたデータの残り

20

22

タであり、UPC2で切り捨てられたアドレスの下位ビットのうち、上位から命令データの最小変化幅、すなわち命令パッファ幅を表わすビットまでがセットされるもので、これを下位プログラムカウンタ（以下LPCと略す）という。第2図に、メモリ上のデータ幅が4バイト、命令パッファ幅が2バイトの場合について、UPC2、LPC3にセットされるアドレスの例を示した。メモリ上のデータ幅は4バイトであるためUPC2には 2^2 ビット以上がセットされ、命令パッファ幅が2バイトであるためLPC3には 2^1 ビットのみがセットされる。このLPC3は、そのカウント値が“0”以外のときは命令パッファ1に、メモリから読み出されたデータのうち命令パッファ幅分が書き込まれることに1減じられ、“0”になると以後は、その“0”を維持するようになっている。4は、メモリから読み出したデータを書き込むべきパッファ部を指示するライトポインタであり、LPC3のカウンタ値が“0”的ときはカウントアップされるが、LPC3のカウンタ値が

"0"以外のときは命令を待ちするようになっている。5は読み出すべきバッファ部を指定するリードポインタであり、命令バッファ1のリード命令が出ることにカウントアップされるものである。6は、命令バッファ1中の2つのバッファ部が空くごとに、これを検出してその旨をUPC2に伝える空きバッファ数検出回路である。7は、メモリに接続されているデータバス8上におけるデータを命令バッファ1の幅に合わせて命令バッファ1に出力するためのセレクタである。9は、ライトポインタ4とリードポインタ5を切り換えるためのセレクタである。尚、10は、命令バッファ1から読み出されたデータを出力するための命令バスである。

このように構成された装置の動作を説明する。ここでは、図3(A)のようにメモリに格納されたデータは4バイト単位で構成され、4バイトごとに読み出されるものとする。また、命令は2バイト単位で変化するものとする。このため、命令バッファ1の幅は2バイトとなっている。ここで例え

ライトポインタ4のカウント値は 1_{10} のため、バッファ部B₁に "b" がセットされる。このときもライトポインタ4はカウントアップされそのカウント値は 2_{10} となる。ここで空きバッファ数検出回路6は、バッファ部B₂とバッファ部B₃が2つ空いていることを検出し、その旨をUPC2に知らせる。このためUPC2はカウントアップされメモリアドレス $4\ 1_{16}$ がセットされる。以下上記と同様の操作が繰り返され、バッファ部B₂、B₃に次々メモリから読み出されたデータ "c"、"d" がセットされる。このようにして、命令バッファ1のバッファ部が2つ空くごとに命令フェッチが繰り返される。

次に、図3(B)の場合を説明すると、最初にUPC2には命令データのアドレス $4\ 2_{16}$ すなわち "01000010" のうち "010000" がセットされLPC3には "1" がセットされる。このためデータバス8には不要データ "x" と命令データ "a" が流れ。これをセレクタ7で上位側 "x" のみとし、ライトポインタ4が示すバッファ部B₀へセットする。

ば1つの命令が、a、b、c、dから成る8バイトであるとすると、それらのメモリ上の配置は、命令の先頭が4バイトの境界に位置する(A)の場合と、4バイトの中間すなわち2バイト目に位置する(B)の場合の2通りが考えられる。また、メモリに格納されたデータは4バイト単位で構成されているため、UPC2には命令アドレスのうち 2^2 ビット以上がセットされ、命令バッファ1の幅が2バイト幅であるためLPC3には 2^1 ビットがセットされることは前述した通りである。

まず、図3(A)の場合を説明すると、UPC2にはメモリアドレス $4\ 0_{16}$ すなわち "01000000" のうち "010000" がセットされLPC3には "0" がセットされる。このアドレスでメモリを読みと、データバス8には "a、b" が流れ。これをセレクタ7で上位側 "a" のみとし、ライトポインタ4が示すバッファ部B₀へセットする。このときLPC3のカウント値は "0" なのでライトポインタ4はカウントアップされ 1_{10} となる。次にセレクタ7により下位側 "b" を選択すると、

B₀へセットする。このとき、LPC3のカウント値は "1" なのでライトポインタ4はカウントアップされず 0_{10} のままであり、同時にLPC3は1減じられ "0" となる。次にセレクタ7により下位側 "a" が選択されると、ライトポインタ4のカウント値は 0_{10} のままであるから再度バッファ部B₀にデータ "a" が書き込まれる。このときバッファ部B₁、B₂、B₃が空いているので、再びフェッチを行えばLPC3は "0" を維持しているから、バッファ部B₁にデータ "b" が、バッファ部B₂にデータ "c" がセットされる。このときバッファ部B₃は空いた状態であるが、連続した2つのバッファ部が空いた状態ではないので、命令フェッチは行われない。この状態はバッファ部B₀にセットされたデータが読み出されて使用されるまで続く。そして、バッファ部B₀のデータが使用されるどそのとき命令バッファ1にセットされたデータはシフトするので連続したバッファ部B₂、B₃が空きとなって命令フェッチが行われ、以上下記と同様の操作がなされ

る。

第3図(A)、(B)に示したように、4バイトの長さでメモリにアクセスすれば命令の先頭がどの位置にあっても、必ず先頭のバッファ部B0に命令データの先頭部分がセットされることになる。

以上は、メモリにアクセスするデータ幅が4バイト、命令バッファの幅が2バイトの場合の説明であるが、命令バッファの幅が、命令の最低変化幅に等しいならば、メモリにアクセスするデータ幅はどれだけあっても通用できるものである。

また、実施例では命令フェッチの場合について説明したが、17位小数点データのように、データ幅の変化が決っているものについては、マイクロ命令でメモリリードを行わなくとも、データの読み込みを行い、このようにしてデータをセットされたバッファから読み出すだけでデータを取り出すことができ、処理の高速化を図ることができる。

【発明の効果】

以上説明したように、本発明によれば、メモリ

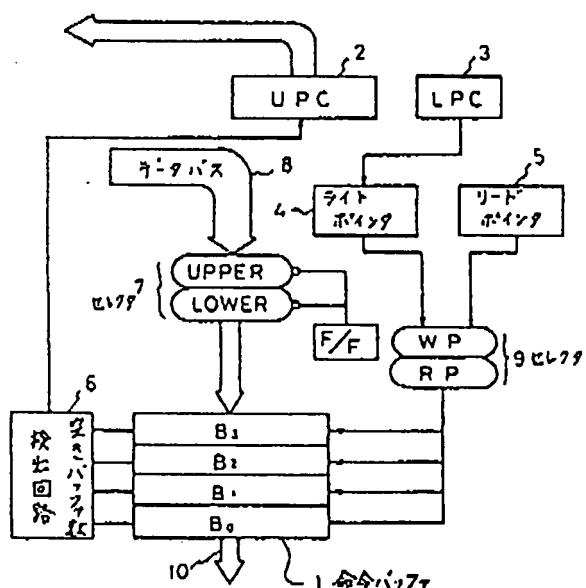
にアクセスできるデータ幅が命令バッファの幅よりも広い場合、命令の先頭位置がどこにあってもメモリ上のデータ幅に合った読み出しを行うだけで、常に命令の先頭を命令バッファの先頭に位置させることができる。

4. 図面の簡単な説明

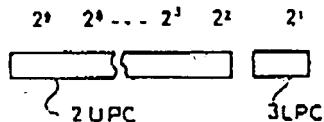
第1図は本発明方式に用いられる装置の構成ブロック図、第2図は第1図に示したUPC、LPCへデータをセットした状態の例を示す図、第3図は第1図に示した装置の動作を説明するためメモリと命令バッファ両方にセットされたデータの状態を示す概念図である。

1…命令バッファ、 2…UPC、
3…LPC、 4…ライトポインタ。

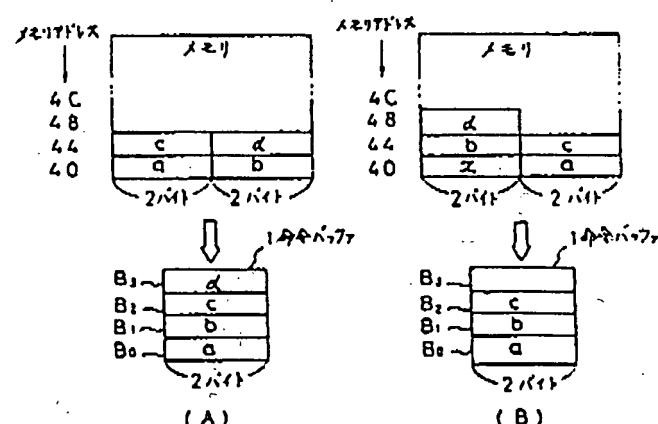
代理人弁理士 本田 勉



第1図



第2図



第3図